DOI: https://doi.org/10.15276/hait.08.2025.20
UDC 004.93

AI-Generated video evaluation by fragment processing

Sergii V. Mashtalir[1)]
ORCID: http://orcid.org/0000-0002-0917-6622; sergii.mashtalir@nure.ua. Scopus Author ID: 36183980100
Dmytro P. Lendel[2)]
ORCID: http://orcid.org/0000-0003-3971-1945; dmytro.lendel@uzhnu.edu.ua. Scopus Author ID: 59390876900
[1)] Kharkiv National University of Radio Electronic, 14, Nauky Ave.Kharkiv, 61166, Ukraine
[2)] Uzhhorod National University, 3, Narodna Square. Uzhhorod, 88000, Ukraine

ABSTRACT

Recent advances in generative AI have led to the development of techniques to generate visually realistic synthetic video. As a result, there is a growing demand for detectors capable of distinguishing between AI-generated videos. In this paper we propose a compact, fragment-based representation of video frames that enables robust spatial-temporal analysis and new approach to discrimination between real and synthetically generated footage. To achieve this goal, each frame is divided into fragments and a square matrix of size *BxB* is formed. Next, we compute the dominant singular value for every fragment, yielding a square S-map. This construction preserves local structure while normalizing geometry, so standard 2-D operators can be applied uniformly. We analyze spatial organization via 2-D Discrete Cosine Transform (DCT) energies and temporal change with robust thresholding to form a binary change mask. Then we apply Connected Component Labeling (CCL) on the binary change mask (4- or 8-connectivity) and compute the area of the Largest Connected Component (LCC). We derive an LCC time series that measures the spatial concentration of change. Empirically, synthetic videos exhibit higher rates of near-binary LCC toggling, longer plateaus, increased mass at rational steps, and fewer unique levels than real videos - signatures consistent with temporal quantization and procedural dynamics. The pipeline is lightweight (fragment-wise rank-1 SVD + CCL on a small grid), auditably interpretable, and suitable for batch screening and edge devices. It attains ROC-AUC ≈ 0.86 and TNR ≈ 0.94 on mixed-resolution datasets with further gains from per-granularity calibration.

Keywords: AI-generated video evaluation; largest connected component; video processing; computer vision; machine learning; combined models; intelligent analysis system; time series analysis

***For citation:* Mashtalir S.V., Lendel D.P. "AI-generated Video Evaluation by fragment processing". *Herald of Advanced Information Technology.* 2025; Vol. 8 No. 3: 316–325. DOI: https://doi.org/10.15276/hait.08.2025.20**

INTRODUCTION AND RELATED PAPERS

Synthetic video generation is progressing very rapidly [1]. The latest models can produce realistic high-resolution videos virtually indistinguishable from real ones. These range from text-prompted approaches such as Stable Video Diffusion [2], VideoCrafter [3], or Sora [4] released by OpenAI, to others such as LumaAI [5] and Gen3AI [6] NeRF-based approaches, which allow synthetic videos to be generated and manipulated based on a set of input images. The emergence of synthetic video generators represents a major technological advancement and a significant escalation in the potential misinformation and disinformation threats caused by generative AI.

AI-generated videos are distributed on social networks and used in advertising production, movies, etc. Sometimes, a criminal may forge a document for entertainment or creative effects. Recognizing whether the video is real or synthetic becomes quite critical. Even in everyday life, while scrolling through the news feed, it would not be bad to understand whether the video is real. In some cases, we can immediately answer the artificial origin of the video based on the content, Fig. 1. We can guess from the content that the video is not real because we know that fish cannot ride bicycles, and rabbits cannot read newspapers while sitting in a cafe.

AI-generated videos based on a scene-changing prepared scenario, with precisely constructed patterns for the specific model, have such a realistic effect that it is practically impossible to distinguish them from the real video with the naked eye.

Many online detectors [7] can detect synthetic video with high efficiency. Most of them use the search for specific patterns or artifacts based on the assumption that a mathematically generated video will contain certain periodicities or inconsistencies not only in the frame but also in the time series. A lot of recent research has focused on detecting artificially generated images [8], [9], [10], [11]. Frame-by-frame processing could take these approaches as a basis. Still, they do not consider interframe changes and inconsistencies in the object's motion and the presence of artifacts [12].

Fig. 1. **Video generated by Sora, LaVie, and Gen2**
***Source*: compiled by the authors**

Recently, many deepfake video detection studies have adopted frequency-based detectors, focusing on the spatial aspects of images through Fourier transforms and filtering [13], [14]. Early detectors identified artifacts in raw videos [15], [16], such as blurred boundaries, color inconsistencies, resolution differences, and flickering. Handling temporal information poses challenges; a typical DeepFake detection pipeline will sample multiple frames from a video, predict per-frame fake probabilities, and then heuristically aggregate these probabilities into an overall fake video probability. This method, however, fails to account for the inherent temporal consistency stemming from real-world constraints, such as stable facial features, unchanged eye colors, and naturally paced blinking. One common way of capturing this temporal information is to use the motion information in videos, commonly represented as Optical Flow [17] (OF). However, one issue is that optical flow estimation requires additional computational resources sequentially, posing a potential efficiency bottleneck.

The survey authors conducted a detailed review of existing AI-Generated Video Evaluation [18] (AIGVE) approaches and identified AIGVE as a distinct research focus on matching AI-generated videos with human perception and instructions. Attempts to combine different approaches encourage researchers to develop tools. One is AIGVE-Tool [19] (AI-Generated Video Evaluation Toolkit), a unified framework that provides a structured and extensible evaluation pipeline for a comprehensive AI-generated video evaluation. AIGVE-Tool integrates multiple evaluation methodologies in novel five-category taxonomy, allowing flexible customization through a modular configuration system.

THE AIM OF THE ARTICLE

We propose a fragment-based representation of video frames- partitioning each frame into a square grid and mapping fragments to a compact σ_1 S-map- to reduce dimensionality while preserving local structure. By tracking inter-frame changes on this S-map and analyzing the normalized Largest Connected Component (LCC) dynamics, we aim to reveal temporal inconsistencies in scene evolution characteristic of synthetic content, enabling practical discrimination between AI-generated and real video.

Our LCC from the binary mask measures spatial concentration of change overtime, a spatiotemporal inconsistency signal related to optical-flow and flicker cues, but lightweight and interpretable.

The aim of the article is to develop are solution-agnostic, fragment-wise pipeline that exposes synthetic temporal inconsistencies while remaining lightweight and auditable for deployment. To achieve this aim, we solve the following tasks: formalize the fragment-wise S-map and the LCC-over-time descriptor; define a compact set of robust decision features and thresholds suitable for mixed resolutions and codec’s; design a simple calibration scheme that yields an interpretable decision signal; evaluate performance on heterogeneous datasets using deployment-oriented metrics (e.g., AUC, TPR/TNR) and cross-validated protocols; analyze computational cost and robustness to grid size, thresholding, and compression artifacts.

APPLICATION OF FRAGMENT PROCESSING FOR THE AI-GENERATED VIDEO

In this section, we will consider the results produced by the developed application. Our experiment used the Kaggle dataset [21], the GenVideo dataset [22], a video surveillance camera, natural video sources, and videos we created using Sora. We treat video as a sequence of frames. Each frame is converted from RGB to a grayscale model so that the value of each pixel carries only intensity information. Thus, problems associated with color rendering and perceptions are excluded from consideration. The Python 3.10.11 application was developed and launched on an Intel Core i5 processor with 16 GB RAM and Windows OS installed to visualize the results of Ky Fan norm usage for video analysis. The application depends on two open-source libraries with Apache license: OpenCV version 4.7.0 and numpy version 1.24.3.

FRAME PARTITIONING AND PER FRAGMENT SINGULAR VALUE DECOMPOSITION (SVD)

Let $X_t \in \mathrm{R}^{H \times W}$ denote the luminance (grayscale) image of video frame t. We partition X_t into a uniform $B \times B$ grid of non-overlapping rectangular blocks $\left\{ X_t^{(i,j)} \right\}_{i,j=1}^{B}$, each of size $h \times \omega$ with $h \approx H / B$, $\omega \approx W / B$.
Videos in datasets have different sizes: [3640x2048], [1920x1080], [1408x768], [1344x768], [1280x720], [854x480], [655x368]. Given a frame of size $H \times W$, we choose the number of fragments per axis B dynamically so that each block has an approximately constant scale across videos (to stabilize Singular Value Decomposition, SVD) and avoids ragged edges when tiling the frame.

Let $h = \left| H / B \right|$ and $\omega = \left| W / B \right|$ be the block height and width. We fix a target side length s (typically 32 px) and a minimum acceptable side s_{min} (typically 16 px). We search over:

$$B = 2, \ldots, B_{max} \quad B_{max} = \min\left(\frac{H}{s_{min}}, \frac{W}{s_{min}} \right).$$

and choose the B minimizing the following cost:

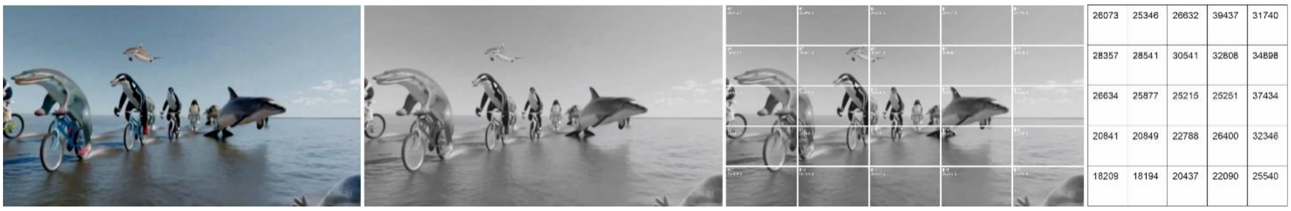
$$\underbrace{\frac{\left| h - s \right| + \left| w - s \right|}{s}}_{\text{block close to target size}} + \underbrace{\lambda_1 \left| \frac{h}{w} - 1 \right|}_{\text{block squareness}} .$$
$$\underbrace{+ \lambda_2 \left(\frac{H \bmod B}{B} + \frac{W \bmod B}{B} \right)}_{\text{edge raggendness}}$$

We typically use $\lambda_1 \approx 0.25, \lambda_2 \approx 0.5$. If two choices tie, we prefer the one with $\min(h, w)$.

For each block we compute the singular value decomposition:

$$X_t^{(i,j)} = U_t^{(i,j)} \Sigma_t^{(i,j)} V_t^{(i,j)^T},$$
$$\Sigma_t^{(i,j)} = diag\left(\sigma_{t,1}^{(i,j)}, \sigma_{t,2}^{(i,j)} \ldots \right).$$

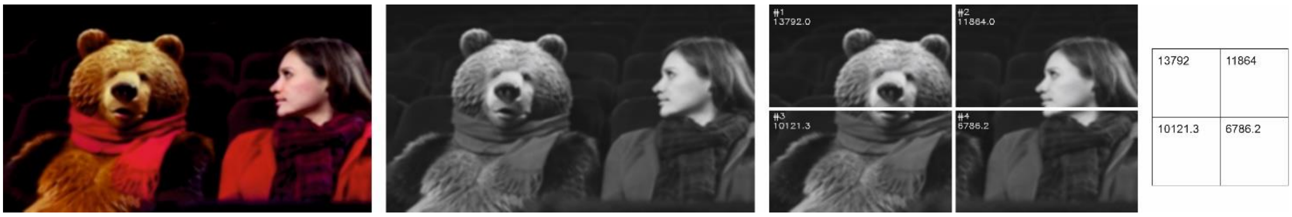
We retain only the dominant singular value $\sigma_{t,1}^{(i,j)}$ (Ky Fan 1-norm, equal to the spectral norm) as a compact scalar descriptor of local structure-contrast. In the research [23], SVD of the matrix and the Ky Fan norm are proposed for scene change analysis. An analysis of the effectiveness of the obtained descriptor across various video data sizes demonstrates that changes in the descriptor for each fragment are independent of both the video resolution and aspect ratio [24]. The rectangle frame has been transformed into a square matrix by SVD, where each element is a Ky Fan 1-norm value used as an object detection descriptor Fig. 2 and Fig.3.



26073	25346	26632	39437	31740
28357	28541	30541	32808	34898
26634	25877	25215	25251	37434
20841	20849	22788	26400	32346
18209	18194	20437	22090	25540

Fig. 2. **Sora generated video. The result of frame-by-frame processing is a new video source in a grayscale model. Frame size is 1280x720. Fragments number is 25**

***Source*: compiled by the authors**



13792	11864
10121.3	6786.2

Fig. 3. **LaVie generated video. The result of frame-by-frame processing is a new video source in a grayscale model. Frame size is 512x320. Fragments number is 4**

***Source*: compiled by the authors**

We chose, for example, real and synthetic videos for better visual demonstration. The videos have the same frame size, 1920x1080, and the same number of fragments – 64. The fluctuations of the Ky Fan norm for the real and synthetic video are shown in the Fig. 4.

CONSTRUCTION OF THE 2-D S-MAP

Stacking the B^2 scalars in row-major order yields the square S-map:

$$S_t \in \mathrm{R}^{B\times B}, \ [S_t]_{i,j} = \sigma_{t,1}^{(i,j)}.$$

Thus each element S_t of corresponds to one spatial fragment of the frame. The mapping converts a rectangular image into a fixed-size square representation on which conventional 2-D operators can be applied efficiently and uniformly across frames.

To capture dynamics, we form the element-wise absolute difference between consecutive S-maps:

$$\vartriangle S_t = \left|S_t - S_{t-1}\right|, t \geq 2.$$

We then produce a binary change mask by robust thresholding:

$$M_t = 1\{\vartriangle S_t > \tau_t\},$$

$$\tau_t = median(\vartriangle S_t) + z * 1.4826 * MAD(\vartriangle S_t),$$

with a typical choice $z \in [2.5, 3.5]$. Then we apply Connected Component Labeling [25] (CCL) on the binary change mask M_t (4- or 8-connectivity) and compute the area of the Largest Connected Component (LCC). We report the normalized LCC:

$$LCC_t = \begin{cases} \dfrac{max_{C\in \mathrm{C}(M^t)} |C|}{\sum M^t} & \sum M^t > 0 \\ 0, & otherwise \end{cases}.$$

Where $\mathrm{C}(M^t)$ denotes 4-connected components on the binary grid and $|C|$ is the area (number of ones). High LCC_t indicates that changes are localized in one dominant region; low values indicate dispersed, flicker-like activity. The sequence $\{LCC_t\}$ over the first T frames constitutes a compact temporal descriptor. A hot map visualization of the progress of the step-by-step LCC calculation is presented in Figure 5.

The dominant singular value of a fragment summarizes its principal contrast-energy while invariant to orthonormal changes of basis within the fragment. Aggregating σ_1 over a grid yields a compact square proxy of the frame's local structure (the S-map). Operating directly on S_t with 2-D tools (DCT) and LCC captures $\vartriangle S_t$, respectively, frequency-spatial organization and the spatial concentration of scene change, producing robust, low-dimensional time series suitable for downstream statistical testing and discrimination.

We analyze fluctuations of the normalized Largest Connected Component (LCC) in the binary change mask (Fig. 6) built from the σ1 S-map because LCC is an interpretable, geometry-agnostic proxy for the spatial concentration of structural change, and it is more robust to pixel-level noise than raw intensities.

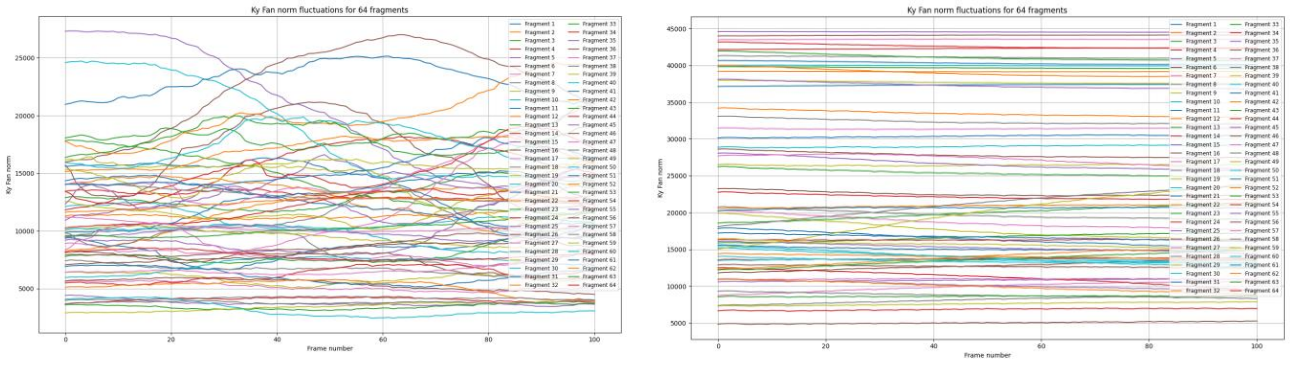


Fig. 4. **Ky Fan norm fluctuation for real and synthetic videos. The videos have the same frame size 1920x1080 and the same number of fragments is 64**

***Source*: compiled by the authors**

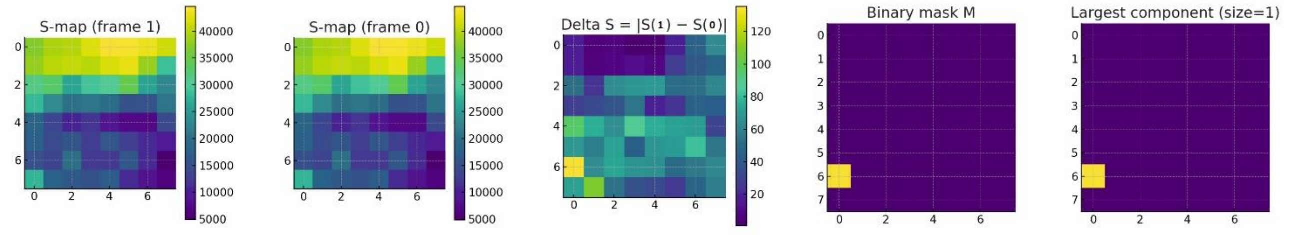


Fig. 5. **Hot map visualization of the progress of the step-by-step LCC calculation**
***Source*: compiled by the authors**

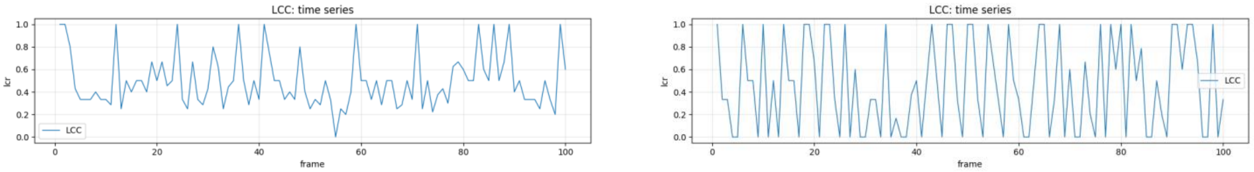


Fig. 6. **Real and synthetic video LCC fluctuation**
Smooth interframe transitions, no interframe “hanging” of the scene, no mirror interframe transitions for real video. Sharp interframe jumps from 0 to 1, indicating spikes, plateaus – “freezing” of the scene, and symmetry in interframe transitions for synthetic video
***Source*: compiled by the authors**

We selected the following descriptors for analysis: extreme low-high jumps, plateau coverage and length, quantization mass near rational levels, lag-1 autocorrelation and Power Spectral Density (PSD) [26] slope, and exceedance shares - capture complementary signatures of temporal regularity and blockwise coherence typical of generative pipelines, enabling discrimination from real footage.

DESCRIPTORS DESCRIPTION

extreme_jumps_0to1 is the number of "nearly binary" low-to-high jumps between adjacent frames:

$$\sum_{t=2}^{N} 1\{x_{t-1} \le zero\_eps \wedge x_t \ge 1 - one\_eps\}$$

where zero_eps =0.08 and one_eps=0.08.

extreme_jumps_1to0 is the number of "nearly binary" high-to-low jumps between adjacent frames:

$$\sum_{t=2}^{N} 1\{x_{t-1} \ge 1 - one\_eps \ \wedge\ x_t \le zero\_eps\},$$

where zero_eps =0.08 and one_eps=0.08.

Plateau count is count of maximal contiguous segments where per-step change stays within tolerance and segment $.length \ge \min\_len$. Segment condition $|x_k| \le delta\_tol$: for all interior steps. Where delta_tol=0.02 and min_len=3.

unique_levels_at_2dec is number of distinct rounded levels occurring at least minimum count per level times. Let $r_t = round(x_t, 2)$, then number of distinct rounded levels:

$$if\ V = \left\{ v : \sum_{t=1}^{N} 1\{r_t = v\} \ge \min\_c\_per\_level \right\} then |V|$$

where min_c_per_level = 2

unique_level_ratio is normalized unique levels: $\text{unique\_levels\_at\_2dec} / \text{N}$.

plateau_fraction_len>=min is fraction of indices covered by plateaus of length ≥ $\min\_len$. If plateau lengths are $l_1 \ldots . l_m$:

$$\frac{\sum_{j=1}^{m} l_j}{N},$$

where $\min\_len = 3$.

value_spike_mass_at_rationals is proportion of frames whose value lies within eps_val of Q :

$$\frac{1}{N}\sum_{t=1}^{N} 1\left\{ \min_{q \in Q} |x_t - q| \le eps\_val \right\},$$

where eps_val=0.02.

step_spike_mass_at_target_steps is proportion of steps whose magnitude lies within eps_step of S

$$\frac{1}{N-1}\sum_{t=2}^{N} 1\left\{ \min_{s \in S} \big| |\triangle x_t| - s \big| \le eps\_step \right\},$$

where eps_step=0.02.

Flicker index is mean absolute frame-to-frame variation

$$\frac{1}{N-1}\sum_{t=2}^{N}\left|\vartriangle x_t\right|$$

acf_lag1 is Lag-1 autocorrelation (with $\bar{x} = \frac{1}{N}\sum x_t$):

$$\frac{\sum_{t=2}^{N}\left(x_t-\bar{x}\right)\left(x_{t-1}-\bar{x}\right)}{\sum_{t=1}^{N}\left(x_t-\bar{x}\right)^2}.$$

psd_slope_lcc is log–log slope of the Power Spectral Density (PSD) over a chosen high-frequency band. Compute:

$$\cdot P\left(f\right)\left(Welch\right), fit \ \log_{10} P\left(f\right) = a + b\log_{10} f; report\, b$$

share_gt_0_2 is fraction of frames with $x_t > 0.2$:

$$\frac{1}{N}\sum_{t=1}^{N}1(x_t > 0.2).$$

share_gt_0_6 is fraction of frames with $x_t > 0.6$:

$$\frac{1}{N}\sum_{t=1}^{N}1(x_t > 0.6).$$

extreme_jump_rate_0to1_or_1to0 is rate of extreme jumps in either direction. Let:

$$C_{01} = \sum_{t=2}^{N}1\left\{x_{t-1} \le zero\_eps \wedge x_t \le 1 - one\_eps\right\}$$

and

$$C_{10} = \sum_{t=2}^{N}1\left\{x_{t-1} \ge 1 - one\_eps \wedge x_t \le zero\_eps\right\},$$

then

$$\frac{C_{01}+C_{10}}{N-1}.$$

plateau_len_mean is mean length of plateaus that satisfy the criteria:

$$\frac{1}{N}\sum_{t=1}^{m}l_j.$$

plateau_len_p90 is 90th percentile of plateau lengths:

$$\text{percentile}_{90}\left(\left\{l_j\right\}\right)$$

We collected descriptors for all videos in the dataset. Example in Table 1.

TP, TN, FP, FN ANALISYS

The dataset was analyzed using a standardized logistic regression within stratified five-fold cross-validation [27], yielding out-of-fold probabilities for every item. Decisions were derived via (i) a global 0.5 threshold and (ii) group-specific thresholds per fragments count selected on each fold's training split to maximize F1 and then applied to the held-out split; performance against the ground truth (positive = Real) was summarized by TP, TN, FP, FN and the derived metrics – accuracy, precision, recall/TPR, specificity/TNR, F1 – and ROC-AUC computed from the out-of-fold probabilities.

We present our results in two tables: Table 2 is the global dataset values of the TP, TN, FP, and FN, and Table 3 is the values of the TP, TN, FP, FN grouped by fragments count**.**

Accuracy is proportion of correctly classified instances:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{N}}.$$

Precision (PPV) is positive predictive value among predicted positives:

$$\text{Presision} = \frac{\text{TP}}{\text{TP}+\text{FP}}.$$

Recall (TPR, Sensitivity) is True-positive rate among actual positives:

$$\text{Recall}\left(\text{TPR}\right) = \frac{\text{TP}}{\text{TP}+\text{FN}}.$$

Specificity (TNR) is True-negative rate among actual negatives:

$$\text{Specificity}\left(\text{TNR}\right) = \frac{\text{TN}}{\text{TN}+\text{FP}}.$$

F1 score is harmonic mean of precision and recall:

$$\text{F1} = \frac{2\text{Precision}\cdot\text{Recall}}{2\text{Precision}+\text{Recall}}.$$

ROC-AUC is area under the ROC curve obtained by varying the decision threshold over the score $p(\text{Real})$. Equivalently, the probability that a randomly chosen positive receives a higher score than a randomly chosen negative:

$$\text{AUC} = \int_0^1 \text{TPR}\left(\text{FPR}\right)\text{d}\left(\text{FPR}\right) = \Pr(\text{s}^+ > \text{s}^-).$$

Table 1. **Part of dataset: Descriptor values for real and AI-generated video**

Descriptors	Frame size is 1920x1080 64 fragments (8x8)		Frame size is 1280x720 25 fragments (5x5)	
	Real Video	AI video	Real Video	AI video
share_gt_0_2	0.969697	0.616162	0.979798	0.30303
share_gt_0_4	0.545455	0.505051	0.727273	0.292929
share_gt_0_6	0.232323	0.343434	0.343434	0.232323
extreme_jump_rate_0to1_or_1to0	0	0.244898	0	0.27551
extreme_jumps_0to1	0	16	0	14
extreme_jumps_1to0	0	8	0	13
plateau_fraction_len>=min	0.060606	0	0.040404	0.626263
plateau_len_mean	3	0	4	5.166667
plateau_len_p90	3	0	4	9.7
plateau_count	2	0	1	12
value_spike_mass_at_rationals	0.747475	0.949495	0.636364	1
step_spike_mass_at_target_steps	0.244898	0.622449	0.142857	0.397959
unique_levels_at_2dec	19	10	23	5
unique_level_ratio	0.191919	0.10101	0.232323	0.050505
flicker_index	1.308911	1.487255	1.187022	1.345981
acf_lag1	0.105166	-0.11014	0.301763	0.099724
psd_slope_lcc	-0.22619	-0.25544	-0.90881	-0.35386

***Source*: compiled by the authors**

Table 2. **Global dataset values of the TP, TN, FP, FN**

TP	TN	FP	FN	Accuracy	Precision	Recall TPR	Specificity NR	F1	ROC AUC
51	136	8	23	0.857798	0.864407	0.689189	0.944444	0.766917	0.85914

***Source*: compiled by the authors**

Table 3 ***The TP, TN, FP, FN values grouped by fragments count***

Frag. count	Files count	TP	TN	FP	FN	Accuracy	Precision	Recall TPR	Specificity TNR	F1	ROC AUC
4	33	0	33	0	0	1			1		
9	3	0	3	0	0	1			1		
16	1	0	1	0	0	1			1		
25	79	27	37	2	13	0.810127	0.931034	0.675	0.948718	0.782609	0.837821
36	38	0	38	0	0	1			1		
64	37	3	23	4	7	0.702703	0.428571	0.3	0.851852	0.352941	0.466667
100	1	0	1	0	0	1			1		
121	4	3	0	0	1	0.75					
256	22	18	0	2	2	0.818182	0.9	0.9	0	0.9	0.325

***Source*: compiled by the authors**

Where s^{+} and s^{-} s^{-} are scores for positive and negative instances, respectively. In our evaluation, scores were the out-of-fold probabilities $p(\mathrm{Real})$ from stratified 5-fold logistic regression; metrics were computed either at a global 0.5 threshold or at group-specific thresholds per “fragments count” selected on the training split.

The datasets source videos have multiple resolutions and aspect ratios. Video data with frame sizes of 1280x720, 1920x1080, and 3840x2160 have a better ratio of real and synthetic videos in quantity, and as a result, the calculated F1 and ROC AUC are more accurate.

CONCLUSIONS

Our approach converts each frame into a compact, square σ_1 S-map via fragment processing, reducing dimensionality while retaining the salient

information about inter-frame structural change. Operating on this low-resolution field and its normalized Largest Connected Component (LCC) dynamics, the method detects temporal inconsistencies characteristic of synthetic video without relying on dense pixel-level motion or heavy end-to-end models. Computationally, it is lightweight: per frame it requires fragmentwise rank-1 SVDs and connected-component labeling on a small $\mathrm{B}\times\mathrm{B}$ grid, yielding near-linear cost in B^2 and modest memory – practical for batch screening or edge deployment.

On the evaluated dataset, the fragment-based pipeline attains OOF ROC-AUC ≈ 0.86 and TNR ≈ 0.94, i.e., a high correct-rejection rate for synthetic videos with few false accepts as “real,” while maintaining interpretability through simple, auditable descriptors (extreme jumps, plateau statistics, quantization mass, ACF/PSD). Per-group thresholding by “fragments count” further improves recall in weaker strata with minimal specificity loss, demonstrating that calibration to spatial granularity is beneficial. Because the representation is resolution-agnostic, it generalizes across native video sizes; nonetheless, performance estimates are most reliable in strata with balanced class ratios (e.g., 1280×720 with 5×5 fragments). Overall, the results support LCC-based fragment analysis as an efficient, interpretable, and effective basis for practical discrimination between synthetic and real video, with headroom for gains via lightweight spatial–spectral add-ons (e.g., DCT energy) and expanded calibration.

DECLARATION ON GENERATIVE AI

The authors have not employed any Generative AI tools.

REFERENCES

1. Tulyakov, S. “Three and a Half Generations of Video Generation Models”. In *Proceedings of the International Conference on Multimedia Retrieval*. 2025, https://www.scopus.com/authid/detail.uri?authorId=57213004407. DOI: https://doi.org/10.1145/3731715.3736185.

2. Blattmann, A., Dockhorn, T., Kulal, S., et al. “Stable video diffusion: Scaling latent video diffusion models to large datasets”. *arXiv*. 2023, https://www.scopus.com/authid/detail.uri?authorId=57212344164. DOI: https://doi.org/10.48550/arXiv.2311.15127.

3. “VideoCrafter: Video generation model repository”. *HuggingFace*. 2023. – Available from: https://huggingface.co/VideoCrafter. – [Accessed: Sept 2024].

4. “Sora: Video generation model”. *OpenAI*. 2024. – Available from: https://openai.com/sora. – [Accessed: Sept 2024].

5. “AI-based video and 3D tools”. *Luma Labs*. – Available from: https://lumalabs.ai. – [Accessed: Sept 2024].

6. “RunwayML: Creative AI tools for video generation”. *Runway*. – Available from: https://runwayml.com. – [Accessed: Sept 2024].

7. “AI Detectors project repository”. GitHub. – Available from: https://github.com/ai-detected/ai-detectors. – [Accessed: Sept 2024].

8. Wang, S. Y., Wang, O., Zhang, R., Owens, A. & Efros, A. A. “CNN-generated images are surprisingly easy to spot... for now”. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2020. p. 8695–8704. DOI: https://doi.org/10.48550/arXiv.1912.11035.

9. Gragnaniello, D., Cozzolino, D., Marra, F., Poggi, G. & Verdoliva, L. “Are GAN generated images easy to detect? A critical analysis of the state-of-the-art”. *arXiv*. 2021, https://www.scopus.com/authid/detail.uri?authorId=55440439700.
DOI: https://doi.org/10.48550/arXiv.2104.02617.

10. Corvi, R., Cozzolino, D., Zingarini, G., Poggi, G., Nagano, K. & Verdoliva, L. "On the detection of synthetic images generated by diffusion models". In *IEEE International Conference on Acoustics, Speech and Signal Processing*. 2023. p. 1–5, https://www.scopus.com/authid/detail.uri?authorId=57216150964. DOI: https://doi.org/10.1109/ICASSP49357.2023.10095167.

11. Wang, Z., Bao, J., Zhou, W., Wang, W., Hu, H., Chen, H. & Li, H. "Dire for diffusion-generated image detection". In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023. p. 22445–22455. DOI: https://doi.org/10.48550/arXiv.2303.09295.

12. Vahdati, D. S., Nguyen, T. D., Azizpour, A. & Stamm, M. C. "Beyond deepfake images: Detecting ai-generated videos". In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2*024. p. 4397–4408. DOI: https://doi.org/10.48550/arXiv.2404.15955.

13. Wang, Y., Peng, C., Liu, D., Wang, N. & Gao, X. "Spatial-temporal frequency forgery clue for video forgery detection in VIS and NIR scenario". *IEEE Transactions on Circuits and Systems for Video Technology*. 2023; 33 (12): 7943–7956. DOI: https://doi.org/10.1109/TCSVT.2023.3281475.

14. Kim, T., Choi, J., Jeong, Y., Noh, H., Yoo, J., Baek, S., & Choi, J. "Beyond Spatial Frequency: Pixel-wise Temporal Frequency-based Deepfake Video Detection". *arXiv*. 2025. DOI: https://doi.org/10.48550/arXiv.2507.02398.

15. Li, L., Bao, J., Zhang, T., Yang, H., Chen, D., Wen, F. & Guo, B. "Face x-ray for more general face forgery detection". In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020. p. 5001–5010. DOI: https://doi.org/10.48550/arXiv.1912.13458.

16. Zheng, Y., Bao, J., Chen, D., Zeng, M. & Wen, F. "Exploring temporal coherence for more general video face forgery detection". In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021. p. 15044–15054. DOI: https://doi.org/10.48550/arXiv.2108.06693.

17. Caldelli, R., Galteri, L., Amerini, I. & Del Bimbo, A. "Optical flow based CNN for detection of unlearnt deepfake manipulations". *Pattern Recognition Letters*. 2021; 146: 31–37, https://www.scopus.com/authid/detail.uri?authorId=6603103996. DOI: https://doi.org/10.1016/j.patrec.2021.03.005.

18. Liu, X., Xiang, X., Li, Z., Wang, Y., Li, Z., Liu, Z. & Zhang, J. "A survey of ai-generated video evaluation". *arXiv*. 2024. DOI: https://doi.org/10.48550/arXiv.2410.19884.

19. Xiang, X., Liu, X., Li, Z., Liu, Z. & Zhang, J. "AIGVE-tool: AI-Generated video evaluation toolkit with multifaceted benchmark". *arXiv*. 2025. DOI: https://doi.org/10.48550/arXiv.2503.14064.

20. Yang, X., Megson, G. M., Tang, Y. Y. & Xing, Y. "Largest connected component of a star graph with faulty vertices". *International Journal of Computer Mathematics*. 2008; 85 (12): 1771–1778. DOI: https://doi.org/10.1080/00207160701619200.

21. "Datasets Repository". *Kaggle*. – Available from: https://www.kaggle.com/datasets – [Accessed: Sept 2024].

22. "Gen-Video Dataset". ModelScope. – Available from: https://modelscope.cn/datasets/cccnju/Gen-Video. – [Accessed: Sept 2024].

23. Koliada, M. "Ky Fan norm application for video segmentation". *Herald of Advanced Information Technology*, 2020; 1 (3): 345–351. DOI: https://doi.org/ 10.15276/hait.01.2020.1.

24. Mashtalir, S. V. & Lendel D. P. "Video fragment processing by Ky Fan norm". *Appl. Asp. Inf. Technol*. 2024; 7 (1): 59–68. https://www.scopus.com/authid/detail.uri?authorId=59390876900s DOI: https://doi.org/10.15276/aait.07.2024.5.

25. Dillencourt, M. B., Samet, H. & Tamminen, M. "A general approach to connected-component labeling for arbitrary image representations". *Journal of the ACM (JACM)*. 1992; 39 (2): 253–280, https://www.scopus.com/authid/detail.uri?authorId=35556471300. DOI: https://doi.org/10.1145/128749.128750.

26. Youngworth, R. N., Gallagher, B. B. & Stamper, B. L. "An overview of power spectral density (PSD) calculations". *Optical Manufacturing and Testing*. VI. 2005; 5869: 206–216. DOI: https://doi.org/10.1117/12.618478.

27. Lei, J. "Cross-validation with confidence". *Journal of the American Statistical Association*. 2020; 115 (532): 1978–1997. DOI: https://doi.org/10.1080/01621459.2019.1672556.

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the research and its results presented in this article

Author Sergii V. Mashtalir is a member of the Editorial Board of this journal. This role had no influence on the peer review process or editorial decision regarding this manuscript

Received 22.07.2025
Received after revision 04.09.2025
Accepted 18.09.2025

DOI: https://doi.org/10.15276/hait.08.2025.20
УДК 004.93

Оцінка відео, згенерованого штучним інтелектом, шляхом фрагментного аналізу

Машталір Сергій Володимирович[1)]
ORCID: http://orcid.org/ 0000-0002-0917-6622; sergii.mashtalir@nure.ua. Scopus Author ID: 36183980100
Лендьел Дмитро Павлович[2)]
ORCID: http://orcid.org/ 0000-0003-3971-1945; dmytro.lendel@uzhnu.edu.ua. Scopus Author ID: 59390876900
[1)] Харківський Національний Університет Радіо Електроніки, проспект Науки 14. Харків, 61166, Україна
[2)] Ужгородський Національний Університет, площа Народна 3. Ужгород, 88000, Україна

АНОТАЦІЯ

Нещодавні досягнення в генеративному штучному інтелекті призвели до розробки методів створення візуально реалістичного синтетичного відео. Як наслідок, зростає попит на детектори, здатні розрізняти відео, згенеровані штучним інтелектом. У цій статті ми пропонуємо компактне, фрагментарне представлення відеокадрів, яке дозволяє проводити надійний просторово-часовий аналіз та новий підхід до розрізнення реальних та синтетично згенерованих відеоматеріалів. Для досягнення цієї мети кожен кадр розділяється на фрагменти та формується квадратна матриця розміром BxB. Далі ми обчислюємо домінантне сингулярне значення для кожного фрагменту, що дає квадратну S-карту. Ця конструкція зберігає локальну структуру, нормалізуючи геометрію, тому стандартні 2D-оператори можна застосовувати рівномірно. Ми аналізуємо просторову організацію за допомогою енергій 2D-дискретного косинусного перетворення (DCT) та часових змін з надійним пороговим регулюванням для формування бінарної маски змін. Потім ми застосовуємо маркування зв'язних компонентів (CCL) до бінарної маски змін (4- або 8-зв'язність) та обчислюємо площу найбільшої зв'язної компоненти (LCC). Ми отримуємо часовий ряд LCC, який вимірює просторову концентрацію змін. Емпірично, синтетичні відео демонструють вищі показники перемикання майже бінарного LCC, довші плато, збільшену масу на раціональних кроках та меншу кількість унікальних рівнів, ніж реальні відео – сигнатури, що узгоджуються з часовим квантуванням та процедурною динамікою. Конвеєр є легким (фрагментно-ранговий SVD + CCL на малій сітці), легко інтерпретується для аудиту та підходить для пакетного скринінгу та периферійних пристроїв. Він досягає ROC-AUC ≈ 0,86 та TNR ≈ 0,94 на наборах даних зі змішаною роздільною здатністю з додатковими перевагами від калібрування на рівні гранулярності.

Ключові слова: оцінка відео; згенерованого штучним інтелектом; найбільший зв'язний компонент; обробка відео; комп'ютерний зір; машинне навчання; комбіновані моделі; інтелектуальна система аналізу; аналіз часових рядів

ABOUT THE AUTHORS

Sergii V. Mashtalir - Doctor of Engineering Science, professor, Informatics Department. Kharkiv National University of Radio Electronics, 14, Nauky Ave, Kharkiv, 61166, Ukraine
ORCID: https://orcid.org/0000-0002-0917-6622. Scopus Author ID: 36183980100; sergii.mashtalir@nure.ua
Research field: Image and Video Processing, Data Analysis

Машталір Сергій Володимирович - доктор технічних наук, професор кафедри Інформатики. Харківський національний університет радіоелектроніки, пр. Науки, 14. Харків, 61166, Україна
ORCID: https://orcid.org/0000-0002-0917-6622, Scopus Author ID: 36183980100 sergii.mashtalir@nure.ua

Dmytro P. Lendel - PhD student. Uzhhorod National University, 3, Narodna Square, Uzhhorod, 88000, Ukraine
ORCID: https://orcid.org/0000-0003-3971-1945; dmytro.lendel@uzhnu.edu.ua. Scopus Author ID: 59390876900
Research field: video processing, video stream segmentation

Лендьел Дмитро Павлович – аспірант. Ужгородський національний університет, пл. Народна 3, Ужгород, 88000, Україна
ORCID: https://orcid.org/0000-0003-3971-1945; Scopus Author ID: 59390876900; dmytro.lendel@uzhnu.edu.ua